

## REMARKS

Claims 1-8, elected as Group I in the Response to the Restriction Requirement submitted November 7, 2003, of which claim 1 is the sole independent claim, are currently pending in the present application. Claims 1-8 stand rejected under 35 U.S.C. § 103(a) as being allegedly  
5 unpatentable over Mattaway et al., U.S. Patent No. 6,185,184 ("Mattaway '184"). In addition, Claims 6-8 stand rejected under 35 U.S.C. § 112, second paragraph. Finally, the claims stand objected to for failing to provide a preferred line numbering format.

After a careful review of the pending claims, Applicants' proposed claim revisions, and the cited references, Applicants respectively request reconsideration in view of the following  
10 remarks.

### **I. APPLICANTS' PRESENTLY CLAIMED INVENTION**

Applicant's presently claimed invention is generally directed to a method for using multiple network addresses for interprocess communication through a common physical layer. For example, and as Applicant's describe with reference to Figure 5, Figure 5 represents a flow  
15 diagram illustrating a method 130 for using multiple network addresses for interprocess communication through a common physical layer.

The method 130 includes creating a first interprocess communication data structure associated with a first network address on a first network device 14 at step 132. The first network device 14 may be the telephony device as illustrated in FIG. 1, although other types of  
20 network devices are possible and the present invention is not limited to the telephony device of FIG. 1. An example of an interprocess communication data structure associated with a network address is a socket that is assigned a new IP 58 address when it is created. A call to a "socket" function from a process may create the first socket as a data structure in memory associated with

the kernel of the first network device 14. For example Socket #1 108 of FIG. 4 may be created by a modified “socket” function call from Process B 84. (Applicants’ Specification p. 26 line 17 – p. 27 line 4).

At step 134, a first communication is established between the first network device 14 and  
5 a second network device 16 using the first interprocess communication data structure and the first network address. For example, the first socket may establish a connection with a target socket on the second network device 16. This may initiate a data flow between the process bound to the first socket and another process on the second network device 16 bound to the target socket. The communication may be connection oriented, such as a TCP 62 virtual  
10 connection, or connectionless, such as a UDP 60 virtual connection. The data flow may follow a modified “bind” or “connect” function call with the “socket descriptor” for the first socket. Process B 84, for example, may initiate a data flow through the data structure of Socket #1 108. The kernel associated Socket #1 108 with its own network interface 114 having the new IP 58 address @X. (Applicants’ Specification p. 27 lines 8 - 17).

15 The first communication passes through the common physical layer for the first network device 14. As illustrated in FIG. 4, the multiple IP 58 addresses at the network layer share a common data-link and device driver 96 for the first network device 14 at its physical layer. Data flows to and from all processes 82-86 through the same physical layer interface 96. Data to or from one process 82-86 may go to one or more socket data structures 108-112. The respective  
20 network layer interface 114 for the first socket 108 encapsulates or decapsulates the data and processes the packets through the common data-link and physical layer 96.

A second interprocess communication data structure is created at step 136. The second interprocess communication data structure is associated with a second network address on the

first network device. The second network address is different from the first network address. For example, Process B 84, having already created Socket #1 108, may also create a second socket, Socket #2 110, by another modified “socket” function call. Socket #2 110 has a different IP 58 address, @Y, compared to Socket #1 108, @X, as the kernel assigns a new IP 58 address when the process makes a modified “socket” function call. There are now two sockets on the first network device 14, and each socket is associated with a different IP 58 address. (Applicants’ Specification p. 27 line 21 – p. 28 line 13).

At step 138, a second communication is established between the first network device 14 and a third network device (not shown) using the second interprocess communication data structure and the second network address. The second socket may then establish a connection with another target socket on a third computer to provide another virtual connection. This may initiate a data flow between the process and a third process on the third network device bound to the target socket. The communication may be connection oriented, such as a TCP 62 virtual connection, or connectionless, such as a UDP 60 virtual connection. The data flow may follow a modified “bind” or “connect” function call with the “socket descriptor” for the second socket. Process B 84, for example, may initiate a data flow through the data structure of Socket #2 110. The kernel associated Socket #2 110 with its own network interface 116 having the new IP 58 address @Y. The respective network layer interface 116 for the second socket 110 encapsulates or decapsulates the data and processes the packets through the common data-link and physical layer 96. In this manner, a host computer, application, process, or other entity may support multiple network addresses and bind each address to a separate socket and/or a separate process. This allows a network device to communicate with other network devices using two or more network addresses. (Applicants’ Specification p. 28 line 14 – p. 28 line 7).

Such a preferred method results in a number of advantages over those systems, like the system that is apparently taught by Mattaway '184, that do not provide for multiple IP address for a single device.

For example, in Applicants' presently claimed invention, more than one IP interface may map to the same physical or logical data-link device. However, each interface will only send and receive data on behalf of a related group of processes as illustrated in FIG. 4. An advantage of the present invention compared to a traditional data-link interface is that it represents an IP address that is bound to a given executing instance of an application (e.g., a process or related group of processes). Such a configuration may be useful for differentiating IP data traffic to or from a particular host based on something other than a transport-layer parameter such as a TCP or UDP port.

For example, in Internet telephony it may be advantageous for each new call to be ascribed a new IP address. Calls are typically IP messages exchanged between telephony devices in a virtual private network. The telephony devices may typically support multiple calls, each call being associated with a process in the telephony device. In this case, the new IP address resides in a private network address space for the virtual private network. NAT tunnels the calls across the (public) Internet. Ascribing a new and unique IP address to each call may ensure that the identity of the caller is hidden as the call traverses the Internet. Each call process should be associated with a respective private IP address. (Applicants' Specification p. 29 lines 8 – p. 30 line 1).

Another advantage of supporting multiple IP addresses is improving switching efficiency. As described above, IP addresses rather than TCP or UDP port number may distinguish traffic for applications. In general, layer 3 (network layer) switching using IP

addresses is expected to be more efficient than layer 4 (transport layer) switching using TCP 62 or UDP 60 ports. Improved switching efficiency may increase the call capacity of the Internet. (Applicants' Specification p. 30 lines 3 – 7).

Yet another advantage of using multiple IP 58 addresses is to differentiate data traffic  
5 associated with different Quality of Service ("QoS"). QoS provides statistical guarantees of throughput, delay, delay variation, and packet loss. QoS is important in the transmission of Internet telephony, multimedia, and video data streams as these transmissions require a guaranteed bandwidth to function. Different applications on the same host computer may need to communicate with a separate QoS. For example, Internet telephony applications may require  
10 a constant or guaranteed bitrate QoS, whereas a web browsing application may require a basic QoS. Differentiating QoS may also improve the ability of the Internet to carry many forms of data communication simultaneously.

Edge routers (24,26) supporting QoS are required to process IP 58 packets differently depending on the QoS of the data communication. Instead of examining the internal details of an  
15 incoming IP 58 packet to determine the appropriate QoS, the edge router (24,26) may simply recognize that an IP 58 address in the header is associated with a particular QoS and process that packet accordingly. As discussed above, layer 3 switching is typically expected to be efficient. Ascribing an IP 58 address to each application may allow the IP 58 address for the application to be associated with the QoS of the application. The IP 58 address and QoS of the application may  
20 be conveyed to the edge router (24,26) to establish a layer 3 switching channel for the application. Therefore, each communication is over a virtual channel between applications associated with a particular QoS and a unique IP 58 address. (Applicants' Specification p. 30 line 8 – p. 31 line 4).

Future communication systems, however, may require a protocol stack capable of supporting multiple IP addresses in a broader sense: different processes on the same host are associated with different IP addresses on the same physical interface. Processes may request a new IP interface with a new IP address rather than share a single IP address with all other processes. Instead of distinguishing traffic to and from processes by port numbers, the traffic may be distinguished by IP address. (Applicants' Specification, p. 3 line 18 – p. 4 line 2).

All of Applicants' presently pending Independent Claims are generally directed to such a method for using multiple network addresses for interprocess communication through a common physical layer. For example, Independent Claim 1 expressly recites "creating a second interprocess communication data structure associated with a second network address on the first network device, wherein the second network address is different from the first network address" and "establishing a first communication between the first network device and a second network device," passing "the first communication [ . . . ] through the common physical layer for the first network device," and then passing a second communication "through the common physical layer for the first network device."

As explained more clearly below, none of the references cited in the February 13, 2004 Office Action teach or suggest, either expressly or inherently, Applicants' method for using multiple network addresses for interprocess communication through a common physical layer.

## **II. CLAIM REJECTIONS UNDER 35 U.S.C. §103(a)**

Claims 1-8 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Mattaway et al., U.S. Patent No. 6,185,184 ("Mattaway '184"). Applicants' respectively traverse.

To establish a *prima facie* case of obviousness based on the combination of the cited references, the cited references must teach or suggest all the claim limitations, and there must be

some suggestion or motivation to modify the references or combine reference teachings. (M.P.E.P. § 2142). The teaching or suggestion to make the claimed combination must be found in the cited references and not in Applicant's disclosure. (M.P.E.P. § 2142).

Applicants' respectively submit that the cited references do not teach or suggest all of the claim limitations of Independent Claim 1. The Mattaway '184 reference simply does not teach or suggest creating "a second network address on the first network device, wherein the second network address is different from the first network address" or the step of passing a first and a second communication "through the common physical layer for the first network device" as presently claimed in Applicants' pending Independent Claim 1.

Rather, Mattaway '184 appears generally directed to what Applicants illustrate in Figure 3 and what Applicants' describe as a "block diagram illustrating a typical stack implementation." (Applicants' Specification p. 6 line 6). (emphasis added). As Applicants describe, this typical stack implementation utilizes multiple sockets for use with only a single IP Address. This appears to be what Mattaway '184 teaches: a single IP address at a network layer.

For example, in Figure 17A, Mattaway '184 discloses a schematic block diagram of a packet transfer sequence between a pair of WebPhone client processes and a global server, "in accordance with the present invention." (Mattaway '184, Col. 22 line 66 – Col. 23 line 2). Mattaway '184 describes that each WebPhone application connects to global server 1500 upon start up to inform global server 1500 that the WebPhone client process is on-line and available to make and/or receive calls. Mattaway '184 states that WebPhone 1536 opens a socket to the global server 1500 and transmits an <ONLINE REQ> packet from WebPhone 1536 to Global server 1500, as illustrated by message 1 and FIG. 17A. (Mattaway '184, Col. 23 lines 2 – 10).

Mattaway '184 then explains that the same Internet Protocol address is used for a second

communication:

5 If the current Internet Protocol address of the callee was returned from global server 1500, the packet transmission sequence illustrated between WebPhones 1536 and 1538 of FIG. 17A transpires. Whether a calling WebPhone knows the Internet Protocol address of the callee WebPhone, as in the case of a fixed Internet Protocol address, or obtains the Internet Protocol address from global server 1500, . . . , the calling sequence to establish a call occurs as follows. WebPhone 1536 opens a socket to WebPhone 1538. Next, WebPhone 1536 transmits to WebPhone 1538 a <CALL> packet as illustrated by message 8 of  
10 FIG. 16A.

(Mettaway '184, Col. 24, lines 29-40) (emphasis added).

In these cited portions of Mattaway '184, there is simply no teaching or suggestion of utilizing a first IP address for a first communication and a second IP address for a second communication and that the second IP address is different from the first IP address. Therefore,  
15 in these cited portions of Mattaway '184, there is simply no teaching or suggestion of the step of passing the first and the second communication "through the common physical layer for the first network device" as presently claimed in Applicants' pending Independent Claim 1.

Rather, Mattaway '184, simply appears to teach away from Applicants' presently claimed invention. In Mattaway '184, there may arguably be two sockets on "a first network device"  
20 (*i.e.*, the WebPhone 1538), but Mattaway '184 does not teach or suggest that each socket is associated with a different IP address. Consequently, Mattaway '184 appears to teach what Applicants' describe as a "typical IP communication process."

As Applicants' explain, typical processes bind to one common IP address. When a process initiates an IP communication, the protocol stack binds the common IP address to the  
25 process. In this sense, all processes, such as the process taught and suggested by Mattaway '184, communicate over the same IP interface. Data traffic to or from one application is typically distinguished from data traffic to or from another application by a transport-layer parameter such



as a TCP or User Datagram Protocol (“UDP”) port. For some processes, however, and as explained by Applicants’, having one common IP address may be restrictive. (Applicants’ Specification, p. 3 line 3 – 8).

Consequently, Applicants’ submit that Mattaway ‘184 does not teach or suggest creating  
5 “a second network address on the first network device, wherein the second network address is different from the first network address.” Naturally, therefore, Mattaway ‘184 does not teach or suggest passing a first and a second communication “through the common physical layer for the first network device” as presently claimed in Applicants’ pending Independent Claim 1.

### **III. CLAIM REJECTIONS UNDER 35 U.S.C. § 112**

10 Claims 6-8 stand rejected under 35 U.S.C. § 112, second paragraph, as the claims recite the limitation “wherein the step.” Independent Claim 1 has been revised to clarify the method steps. Applicants respectively request that this objection be withdrawn.

### **IV. CLAIM OBJECTIONS**

The claims stand objected to as they do not contain a preferred line number format. The  
15 line numbers for the claims have been modified to reflect the preferred format. Applicants respectively request that this objection be withdrawn.

### **V. SUMMARY**

In conclusion, it is respectively submitted that Applicants have overcome each of the Examiner’s rejections over the cited references.

20 It is submitted, therefore, that all pending Claims 1 - 8 are in condition for allowance and early notice to this effect is earnestly solicited. Independent Claim 1 is allowable for the reasons stated above. The remaining pending claims 2-8 are dependent on allowable Independent Claim

1 and are therefore allowable for at least the reasons discussed with regard to Independent Claim 1.

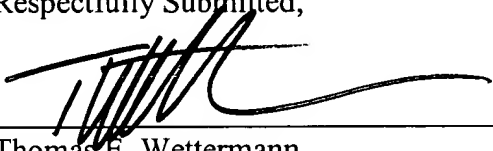
If there are any additional matters which may be resolved through a telephone interview, the Examiner is respectfully requested to contact Applicant's undersigned representative.

5

Respectfully Submitted,

Date: 5/13/04

By:

  
Thomas E. Wettermann  
Reg. No. 41,523

10